**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МОЭВМ**

отчет

**по лабораторной работе №3**

**по дисциплине «Операционные системы»**

тема: **Исследование структур загрузочных модулей**

|  |  |  |
| --- | --- | --- |
| Студент гр. 6382 |  | Вайгачёв А.О. |
| Преподаватель |  | Губкин А.Ф. |

Санкт-Петербург

2018

**Цель работы:** Для исследования организации управления памятью необходимо ориентироваться на тип основной памяти, реализованный в компьютере и способ организации, принятых в ОС. В лабораторной работе рассматривается не страничная память и способ управления динамическими разделами. Для реализации управления памятью в этом случае строится список занятых и свободных участков памяти. Функции ядра, обеспечивающие управление основной памятью, просматривают и преобразуют этот список.

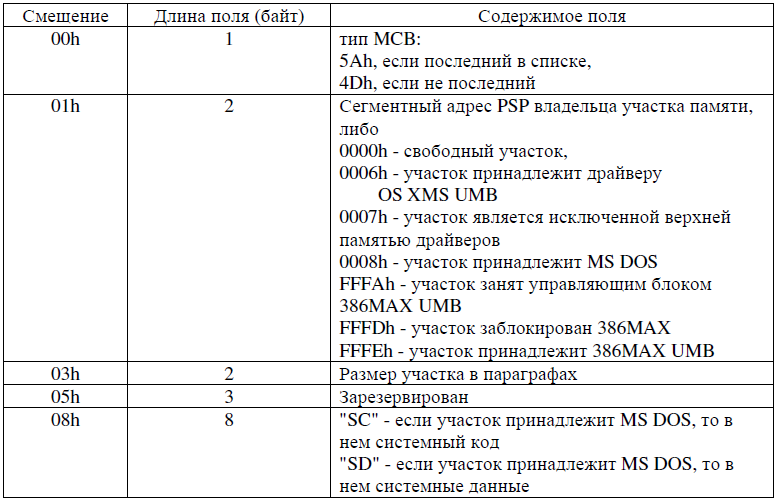
**Порядок выполнения работы:**

1. Написать и отладить программный модуль типа .COM, который выбирает и распечатывает следующую информацию:
   1. Количество доступной памяти
   2. Размер расширенной памяти
   3. Выводит цепочку блоков управлению памятью.
2. Изменить программу так, чтобы она освобождала память, которую она не занимает.
3. Изменить программу так, чтоб после освобождения памяти, программа запрашивала ещё 64 КБайта.
4. Изменить программу так, чтоб программа запрашивала 64 КБайта до освобождения лишней памяти.

**Необходимые сведения для составления программы:**

Учет занятой свободной памяти ведется при помощи списка блоков управления памятью MCB(Memory Control Block). MCB занимает 16 байт (параграф) и располагается всегда с адреса кратного 16(адрес сегмента ОП) и находится в адресном пространстве непосредственно перед тем участком памяти, которым он управляет.

MCB имеет следующую структуру:



По сегментному адресу и размеру участка памяти, контролируемого этим MCB можно определить местоположение следующего MCB в списке.

Адрес первого MCB хранится во внутренней структуре MS DOS, называемой “List of Lists” (список списков). Доступ к указателю на эту структуру можно получить исползуя функцию 52h “Get List of Lists” int 21h. в результате выполнения этой функции ES:BX будет указывать на список списков. Слово по адресу ES:[BX-2] и есть адрес самого первого MCB.

Размер расширенной памяти находится в ячейках 30h, 31h CMOS. CMOS это энергонезависимая память, в которой хранится информация о конфигурации ПЭВМ. Объем памяти составляет 64 байта. Размер расширенной памяти в Кбайтах можно определить обращаясь к ячейкам CMOS следующим образом:

Mov AL,30h ; запись адреса ячейки CMOS

Out 70h,AL

In AL,71h ; чтение младшего байта

Mov BL,AL ; размера расширенной памяти

Mov AL,31h ; запись адреса ячейки CMOS

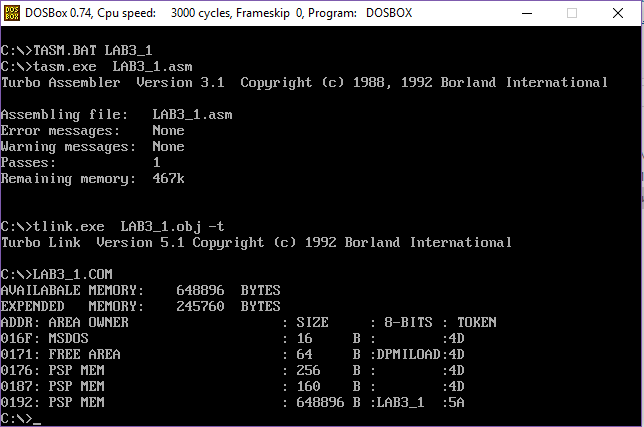
Out 70h,AL

In AL,71h ; чтение старшего байта

; размера расширенной памяти

**Работа программы**

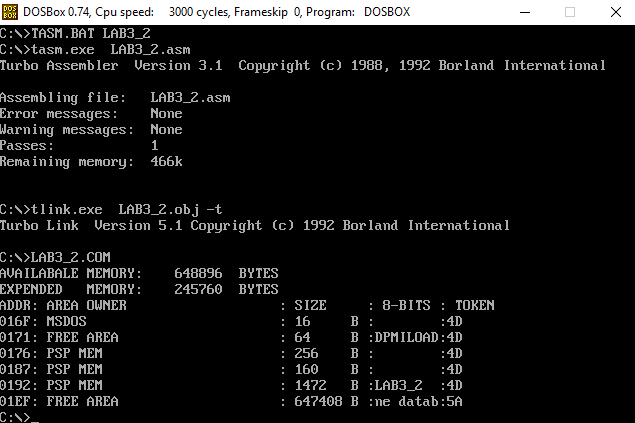
1)lab3\_1.com



Видим, что память состоит из 5 блоков памяти. Самый большой занимает наша программа.

2)lab3\_2.com

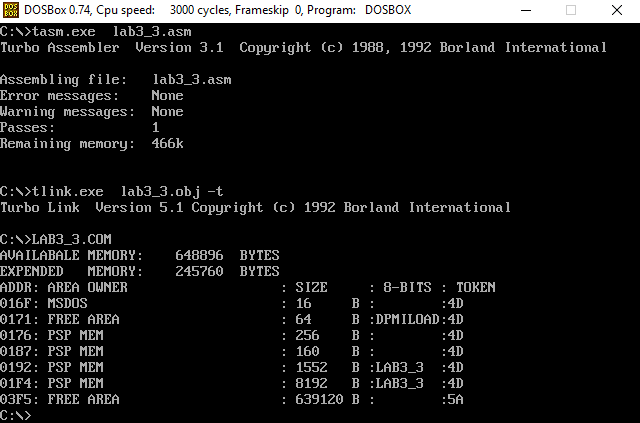
Освобождаем память



Видим, что появился блок свободной памяти. Наша программ компактно умещается в блоке размером 1472 б.

3)lab3\_3.com

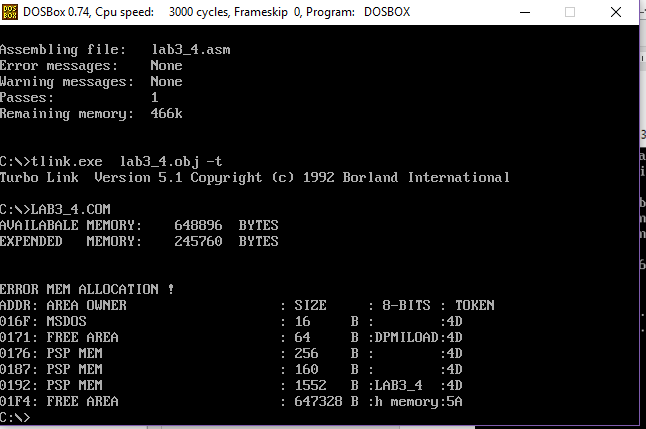
Запрашиваем память



Видим, что создано два блока нашей программы.

4)lab3\_4.com

Сначала запрашиваем память, потом переорганизовываем память



Что-то пошло не так. Видимо можно запрашивать память только тогда, когда есть свободная. Вот так сюрприз.

**Вывод:** В ходе данной лабораторной работы было изучено: нестраничная память, способ управления динамическими разделами, список занятых и свободных участков памяти, функции ядра, обеспечивающие управление основной памятью. Также была исследована работа функций управления с основной памятью.

**Ответы на вопросы:**

1. **Что означает «доступной объём памяти»?**

**Ответ:** Это объем памяти, которую может использовать программа для хранения своих данных.

1. **Где MCB блок вашей программы в списке?**

**Ответ:** в 1, 2, 4 пунктах это блок по адресу **0192H**, в 3 пункте это блоки по адресам **0192H** и **01F4H**.

1. Какой размер памяти занимает программа в каждом случае?

**Ответ:**

№ 1: 648896 байт.

№ 2: 1472 байт.

№ 3:1552 + 8192 = 9744 байт

№ 4: 1552 байт.